

LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

LA-UR-85-129

CONF-850598-1

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

LA-UR--85-129

DE85 006076

TITLE: COPING WITH COMPUTER TERMINOLOGY

AUTHOR(S): Kyle Wheeler

NOTICE
COPIES OF THIS REPORT ARE AVAILABLE
has been reproduced from the best
available copy to permit the broadest
possible availability.

SUBMITTED TO: 32nd Annual International Technical Communicators Conference
(ITCC) for publication in the conference proceedings, May 19-23,
1985, Houston, Texas

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

By acceptance of this article the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution or to allow others to do so, for U.S. Government purposes.

The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy.

Los Alamos Los Alamos National Laboratory
Los Alamos, New Mexico 87545



Kyle Wheeler*
Technical Writer/Editor
Computing and Communications Division
Los Alamos National Laboratory
Los Alamos, New Mexico 84548
505-887-7038

I work in the world of computer terminology. Every day I encounter words that aren't in any dictionary or words that sometimes are used as two words, sometimes are hyphenated when used as a unit modifier, and sometimes are written as one word. Keeping track of when to use one word, two words, or a hyphenated word is a time-consuming task, especially because there is no one way to do it. The authors I work with send their papers to a variety of journals, and many journals have a *preferred way* of spelling.

Another problem I deal with frequently is what to do about acronyms. Some people use FORTRAN, and others use Fortran; both are acceptable. There is no way to avoid acronyms, and it seems that new ones are created every day.

To maintain my sanity and a professional rapport with the many authors I work with, I decided to develop some ways to cope with the ever-changing computer terminology. I describe here the methods that I find useful (including using the computer, which is a VAX running the UNIX operating system). You need not have access to fancy or expensive equipment though to modify my methods for use in another environment.

INTRODUCTION

Imagine how I felt in the following situation. I had edited an author's paper and changed every occurrence of *online documentation* to *on-line documentation*. I explained to him that the style guide for the publisher of his paper preferred this use. That made perfect sense to him, so he started using *on-line* as an adjective and *on line* as an adverb in his papers, letters, and memos. Not too long after that, a memo of his came back from one of his supervisors with instructions to change all of the *on-line* adjectives and *on line* adverbs to *online*. He was confused, and he came to talk with me about it.

It occurred to me that the words you choose, and even the spelling of the words you choose, depend on the intended audience of the paper, letter, or memo, and that many of the people who were generating all these words didn't understand that there is *more than one right way* to write. I decided that as the person these people rely on for editorial advice, I better devise some ways to help them (and help myself too!).

Many of you may already be doing something similar to the things I describe here. Terminology is changing so rapidly that I don't think any of us can keep up, and we all must devise methods of coping.

ESTABLISH A GLOSSARY OF COMMON TERMINOLOGY

From my previous experience, I knew that a style guide is a great aid to technical editors. It seemed even more necessary to me as a newcomer in the field of computing—I needed to learn the common terms, acronyms, and abbreviations. I needed to take whatever information was already available and customize it until I found it useful.

Spelling Variations and Hyphenations

When I started working in my current job, I was given a list of terms and acronyms commonly seen in papers, reports, and correspondence in the Computing and Communications (C) Division. Many people in the division used that list as a mini style guide. At first, some of the words on that list were abhorrent to me (for example, *online*, *signon*, *logon*, and *login*). I couldn't find them in my *American Heritage Dictionary* or any of my other reference books, but I kept seeing them published in journals, magazines, and newspapers. I didn't want to change *log-in procedure* to *login procedure* because the mini style guide listed that spelling. It really was a confusing situation for authors and for me. I realized that the best thing to do was to keep a list of all of the acceptable spellings instead of trying to follow one prescribed way.

So I started rewriting my list. Where it listed *queuing* I added *queueing*. I couldn't find *queueing* in any of my three dictionaries, but an author brought me a page copied out of her dictionary showing *queueing* as an acceptable spelling. Where *database* was listed, I added *data base* and *data-base*. I noted the word use of each spelling: *data base* (noun); *data-base* (adj); *database* (noun, adj).

I think people are confused by the situation where two words are used for an adverb and the same two words are hyphenated when used as an adjective. For example,

- Our goal is to make *on-line* documentation available for all systems.
- The documentation is available *on line*.

My theory is that most computer scientists don't know the difference between an adjective and an adverb (and don't care) and they don't understand the differences between the sentences above. If you leave the editing to them, they will either hyphenate every occurrence of *on line* or they will make it one word whenever they see it. That certainly is simpler and it is consistent. And I think it's practical.

Another problem that confuses many people occurs when a string of words is hyphenated when it comes before a noun but that same string of words is not hyphenated when it comes after a noun. This rule is confusing because there are so many exceptions to it. Consider the following correct sentences.

This is the state-of-the-art computer.

This computer represents the state of the art.

The task of writing documentation is never-ending.

Writing documentation is a never-ending task.

The pipe must be 10 m long

We will need a 10-m-long pipe to replace the broken one.

Some people act as if I'm making up the rules as I go along when I explain the difference. Although I'm the editor and I like to think of myself as a careful writer, even I think some of the rules aren't very practical.

* Work performed under the auspices of the U. S. Department of Energy

In the field of computing, the trend seems to be to make one word out of two words. I am sometimes tempted to follow that trend. I have trouble remembering all the different spellings and I have to look up the same words repeatedly, so it's a real temptation to make them all one word. (I think I could tolerate runtime, but what about cross-section?) I find it helpful to add those "problem" words to my list as well.

By adding acceptable spellings and hyphenations to my list and by explaining them a little bit, I was doing myself a favor. Perhaps the people whose papers I edit will understand that I know what I'm doing when I use *data base* in one of their papers and *database* in another (depending on the style of the publisher). When the author isn't sure who will publish the paper, I use the spelling that the author prefers, making sure that the spelling is consistent throughout the paper. By the way, no list can be complete—not when terminology changes so frequently. I try to keep the most commonly used spellings and hyphenations on my list. This list isn't supposed to replace my reference books—its function is to put common problems in one small document.

Acronyms

Another thing I found helpful, especially because the field of computing was new to me, was to add the most commonly used acronyms and their definitions to my list. Acronyms are very common in computing. I don't believe acronyms can be avoided, but I do think their use can be limited to only the most important ones. Why not spell out scientific workstation every time it occurs rather than use SWS? Doing this really makes a document easier to read. I believe acronyms should be defined, and writers should remember what they mean; it is redundant to use *GKS System* when GKS is an acronym for Graphics Kernel System. Also, I do not favor using odd capitalizations for words in the acronym's definition, although this is hard for some people to accept. For example, I often see

extended NETWORK Access System (XNET).

I prefer to use Extended Network Access System (XNET). And if an acronym doesn't refer to a specific place or thing, I don't capitalize any of the words: computer-based terminal (CBT).

In major documents that use a lot of acronyms, I always add a glossary. People don't necessarily read large documents from start to finish, so they may have missed the definition of an acronym. To generate part of the glossary, I run my files through the spelling package on the UNIX system that I use. It regards acronyms as misspelled words; it even lists them for me in alphabetical order. All I have to do is define them.

Keeping track of acronyms serves another useful purpose. I can never remember when I see terms spelled with all capital letters whether they are in fact acronyms, so by referring to my list, I can quickly find out. A mistake I frequently see is ADA: many people think Ada is an acronym, but it is the name of a programming language and it was taken from a woman's first name. This is discussed in more detail below.

Upper Case, Lower Case, Bold, and Italic

Many people who write and edit where I work use the UNIX operating system, which means it is easy to use bold, italics, and upper and lower case. Authors often tell me that they want certain words to stand out, so they put them in upper case, or they put them in bold or italic, or they do all three! I try to convince them that a page liberally sprinkled with every font available defeats the purpose of making anything stand out. However, I mention this topic because I think it adds to the confusion of the readers (and the writers) about spelling. For example, many people use all capital letters for the names of systems, software, and utilities. Thus, I often see combinations of TELNET and Telenet and ARPANET and Arpanet in one document. In general, words that are acronyms should be in all caps, and words that are not acronyms but merely names should be treated as any other name.

Abbreviations

There are a few abbreviations that I had never encountered before I started editing computing papers, so I have them on my list too. A few of these include bit/s (bits per second), byte/s (bytes per second), capital K when the number refers to memory (64K memory), but lower case k for kilo (thousand), as in 9600 kbit/s.

Quotation Marks and Other Punctuation

Have you ever read anything related to computing where the quotation marks were in the correct position? The established convention is to place periods and commas within quotation marks. Yet I almost never see them there. Computer scientists put the quotation marks where they belong logically. I have to do it all the time, too. For example, to get the title of a reference but not the comma following it to come out in italic font, I have to type in the following:

"Name of Book".

In most papers and reports, I follow established conventions for quotation and other punctuation marks, but computer documentation is another matter. How would you punctuate the sentence that follows?

To change to another person's file space and look at the files in their 2yrplan directory, type in the following command:

```
cd /u0/ktw/2yrplan
```

Most editors will put a period at the end because all sentences should end with a period. Most documentation writers will leave the period out because they don't want anyone to think that the period is part of the command. Personally, I think commands like the one above should be treated like equations, where a period is put two spaces after the end of the command.

KEEP A SMALL BUT SEPARATE LIST FOR EVERY EDITING JOB

How do I remember to use *data base* and *data-base* in the paper I edit on Monday and *database* in the paper I edit on Friday? I wouldn't be able to do it if I didn't keep a list for every editing job that I do. When I start a job, I list, in alphabetical order, all the acronyms, strange spellings, and hyphenations that I use. When I have to put a job aside for several days, I can pick it up again and continue with my editing. Without that list, I wouldn't be able to remember what I had done. I would have to search through the file looking for those words to see what I had done before I could go on. When you have access to a computer or a word processor, that isn't difficult to do, but without those tools, you have to go through it yourself, page by page—a slow and tedious task. Even with access to computers and word processors, I save time by generating these lists. When I complete a job, I file the list along with a copy of the completed paper just in case it comes back someday to be revised.

MAINTAIN A BOOKSHELF OF GOOD REFERENCE BOOKS

I don't think I could function for two days as a technical editor without a few good reference books within easy reach. No one can remember all the spelling and grammar rules that an editor has to rely on to do the job. Authors are much more willing to accept an editor's word if the editor can find a source in a published book. I am always thumbing through these books and would be lost without them. So I recommend that you assemble a collection of books, become familiar with their contents so you can use them quickly and easily, and if you are lucky enough to have a slow afternoon, spend some time reading them. The bibliography at the end of this paper lists some of my favorites, with the more general books listed first and the books specific to computing listed second. Try not to think of your reference books as bibles. They are there to guide you, but they can often frustrate you because they sometimes contradict each other. Use them to guide you, not to rule you.

LET YOUR TERMINAL DO SOME OF THE WORK

Throughout this paper I have touched upon ways that I use my terminal to help me with my work. Using a terminal can be a real timesaver, but I don't depend on it too much because I know that I will be held accountable for all the mistakes the terminal didn't find. I would like to summarize some of the the pros and cons here.

Take advantage of any tools that you have available on the system. If you have spell, diction, or Writer's Workbench, use it. Add variant spellings to your dictionary if you can.

Use the spelling package if you have one to generate a list of acronyms. The system that I use thinks acronyms are misspelled words, so it lists them in alphabetical order for me. I go through the list, add the definitions, and I have part of my glossary.

Use the search command to check spelling. For example, if I want to check to see if database is always spelled as one word, I have the system search for every occurrence of ata. It will then find all of the following: data, data-, database, Data, Data-, and Database. As I go through, I can check each word to make sure it is correct. Beware using a global command to find or change things; for example, if you use a global command to change all occurrences of data base to database, you might end up with a few embarrassing occurrences of Data base or data-base. It's a good idea to search through a file from front to back, using human judgment to decide whether a change is appropriate.

There is another thing I have to watch out for when I use the search command. If I want the system to search for something that has two words, for example, second order, and if I type in those two words, it will not find those occasions where the first word falls at the end of a line and the second word begins a new line. I learned that by trial and error. So now I give the system as few characters to search for as is possible. That means it takes more time to search through a file, but I can feel pretty sure that the system found each occurrence of whatever it was searching for.

CONCLUSIONS

As editors, should we defend our dictionaries and usage guides and ignore what is happening in the field? Should we pick one way to spell words that have more than one acceptable spelling and always use that spelling? Would we provide our authors a service by doing so? I don't think so. Although I don't particularly like the verb *access*, it is used frequently, and I would be fighting a losing battle trying to stop it. There are better battles to fight. But I certainly do not advocate throwing away our dictionaries and style guides. (I wouldn't mind tossing out all the reference books that contradict one another, but then I wouldn't have anything left on my shelf.) I believe in being a careful writer—not a nit picker, not a rigid defender of all the rules, but a person who is familiar with the rules and who can decide when the rules are being too restrictive.

Is there any way that you can use my methods to help you do your job? I think so. Before I started editing papers about computing, I edited papers on a variety of technical topics. No matter what field you are in, certain words are used over and over—in geology it may be *groundwater* and *ground water*. The same problems appear—it's just the terms that are different. Perhaps it really doesn't matter whether we use one word or two or whether strings of words are hyphenated before or after a noun—as long as we are clear in delivering our ideas and consistent in our use of words and style within one document.

ACKNOWLEDGMENTS

I would like to thank Margery McCormick, who was the editorial expert in C Division before I came along and whose fine standards I have tried to maintain. It was she who generated the original list for my glossary. I would also like to thank the staff of the Computer Documentation Group for their support and constructive criticism. I am grateful to Larry McFarland, Almera Turner, and Elaine Stanlick for their guidance during my first few years as a technical editor.

BIBLIOGRAPHY

General Reference Books

The American Heritage Dictionary of the English Language, William Morris, Editor, Houghton Mifflin Company, Boston (1981).

Webster's Third New International Dictionary of the English Language: Unabridged, Philip Babcock Gove, Ph.D., Editor in Chief, G. & C. Merriam Company, Springfield, Massachusetts (1971).

Dictionary of Scientific and Technical Terms, Daniel N. Lapedes, Editor in Chief, McGraw-Hill Book Company (1974).

Floyd C. Watkins and William B. Dillingham, *Practical English Handbook*, Sixth Edition, Houghton Mifflin Company, Boston (1982).^{*}

Kaethe Ellis, compiler, *The Word Book II*, Houghton Mifflin Company, Boston (1983).

William Strunk Jr. and E. B. White, *The Elements of Style*, Third Edition, Macmillan Publishing Co., Inc., New York (1979).

Theodore M. Bernstein, *The Careful Writer, A Modern Guide to English Usage*, Atheneum, New York (1982).

Technical Reference Books

Charles J. Sippl and Roger J. Sippl, *Computer Dictionary and Handbook*, Third Edition, Howard W. Sams & Co., Inc., Indianapolis, Indiana (1983).

Ellen Swanson, *Mathematics into Type*, American Mathematical Society, Providence, Rhode Island, Revised edition (1979).

Donald Spencer, *Computer Dictionary for Everyone*, Charles Scribner's Sons, New York, 1979.

^{*}These books are really handy because they are so small.